



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

The online pivot

Citation for published version:

Alex, B, Llewellyn, C, Orzechowski, P & Boutchkova, M 2021, The online pivot: Lessons learned from teaching a text and data mining course in lockdown, enhancing online teaching with pair programming and digital badges. in D Jurgens, V Kolhatkar, L Li, M Mieskes & T Pedersen (eds), *Proceedings of the Fifth Workshop on Teaching NLP*. Association for Computational Linguistics, pp. 138-148.
<<https://www.aclweb.org/anthology/2021.teachingnlp-1.24/>>

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Publisher's PDF, also known as Version of record

Published In:

Proceedings of the Fifth Workshop on Teaching NLP

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



The Online Pivot: Lessons Learned from Teaching a Text and Data Mining Course in Lockdown, Enhancing online Teaching with Pair Programming and Digital Badges

Beatrice Alex

Literatures, Languages and Cultures
University of Edinburgh
Edinburgh
Scotland, UK
balex@ed.ac.uk

Clare Llewellyn

Social and Political Science
University of Edinburgh
Edinburgh
Scotland, UK
clare.llewellyn@ed.ac.uk

Pawel Michal Orzechowski

University of Edinburgh Business School
University of Edinburgh
Scotland, UK

Maria Boutchkova

Pawel.Orzechowski@ed.ac.uk Maria.Boutchkova@ed.ac.uk

Abstract

In this paper we provide an account of how we ported a text and data mining course online in summer 2020 as a result of the COVID-19 pandemic and how we improved it in a second pilot run. We describe the course, how we adapted it over the two pilot runs and what teaching techniques we used to improve students' learning and community building online. We also provide information on the relentless feedback collected during the course which helped us to adapt our teaching from one session to the next and one pilot to the next. We discuss the lessons learned and promote the use of innovative teaching techniques applied to the digital such as digital badges and pair programming in break-out rooms for teaching Natural Language Processing courses to beginners and students with different backgrounds.

1 Introduction

It was spring 2020 and it felt like we were in crisis mode. We wanted to teach a text and data mining (TDM) pilot course but because of social distancing measures we could not do it in a physical classroom. We had to learn new ways of interacting online and using a multitude of different technologies and we needed to do it fast. We had been planning this course for a while before Covid-19 hit. We were designing a TDM for Humanities and Social Science students but because of the situation we had to adapt the way we delivered it. Rather than hybrid teaching as intended, accommodating in-classroom,

online synchronous and online asynchronous students, we had to fully commit to online methods in a matter of a few weeks. We decided to plunge headlong into the digital teaching world.

The easiest way would have been to post videos of a traditional style lectures – it is very tempting to take this approach. We felt, however, that it was important that we maintained what is good about teaching when everyone is in the same room, the collaboration, its social aspects, the feedback, all of which you lose when a student sits on their own in a room watching a pre-recorded lecture.

We decided to run a TDM boot camp to virtually test our new course which we were planning as part of the Edinburgh Futures Institute (EFI) postgraduate programme.¹ We wanted to not only teach fundamental methods for text mining corpora to programming novices but also teach ourselves how to become better practitioners in teaching in an online world.

In this paper, we will describe our methods and experience for porting an in-person TDM course into the online world. In the next section we will present related publications on teaching Natural Language Processing or TDM courses. We then describe the academic backgrounds of the teaching team (Section 3.1) and provide an overview of our course (Section 3.2). Sections 3.3 and 3.4 explain how we taught and adapted it in two online pilot runs delivered in June and September 2020. We

¹EFI is a new institute at the University of Edinburgh which will support interdisciplinary research and teaching for the whole institution. <https://efi.ed.ac.uk>

provide information on how we collected relentless feedback during and after each course and include a detailed account of one participant of the first pilot and how it has affected her teaching (Section 4). Finally, we summarise what we learned from these experiences (Section 5) and lay out future plans for our TDM course (Section 6).

2 Related Work

There are two aspects that we consider of importance in relation to this work, the course content, Natural Language Processing (NLP), and the environment, teaching online during a pandemic. In this section we explore both topics.

NLP educators choose which aspects to teach based on multiple constraints such as class length, student experience, recent advancements, program focus, and even personal interest.

Our TDM course is fundamentally designed to be cross-disciplinary as we are teaching NLP and coding to students from multiple schools and backgrounds including linguistics, social sciences and business. [Jurgens and Li \(2018\)](#) point out that NLP courses are designed to reflect, amongst other things, the background and experience of the students. [Agarwal \(2013\)](#) explains that in courses such as these the majority of students, who he calls “newbies in Computer Science”, have never programmed before. He highlights that we can increase experience through homework tasks which we did both before the course and in-between each session. [Hearst \(2005\)](#) states that in these circumstances it is not important to place too much emphasis on the theoretical underpinnings of NLP but to focus on providing instructions for students on what is possible and how they can use it on their own in the future. We based our approach on using the NLTK² and spaCy³ Python libraries as well as used examples inspired by [Bird et al. \(2009, 2005\)](#). We aim to explain how text analysis works step-by-step using clear and simple examples. We thereby aspire to develop and broaden humanities and social science students’ data-driven training and give them an understanding of how things work inside the box, something for which there is still a significant need in their core disciplines ([McGillivray et al., 2020](#)).

Teaching text analysis to non-computer scientists has been explored in texts such as [Hovy \(2020\)](#). For

our course we had to consider the variety of backgrounds and experiences that this would encompass and needed to use a pre-course learning task and office hours to provide a more level knowledge starting point. We also had to design the course to keep more advanced students engaged while not intimidating learners who may find it more challenging. We used core material to explain principle concepts (such as tokens, tokenisation, and part-of-speech (POS) tagging etc.) but with a hands-on approach. We avoided too much technical detail and put the material in the context of projects we have worked on ourselves to demonstrate how each analysis step becomes useful in practice.

As we taught our TDM course online in the context of a worldwide pandemic, we also report on related work in the area of online teaching, and with respect to the challenges in which we are teaching. Massive open online courses (MOOC) generally focus on providing online access to learning resources to a large number and wide range of participants. This has led to a desire to automate teaching and innovate digital interaction techniques in order to engage with large numbers of students. Whilst our intention was to teach a limited number of students, we hoped to use and draw upon innovation in this area in order to improve the experience for our students. E-learning and technology should not be seen as an attempt to replace or automate human teaching, although this can often be a fear articulated by teachers. In a discussion of automation within teaching [Bayne \(2015\)](#) argues that we can design online teaching and still place human communication at the centre with technology enhancing the learning of the student. Bayne suggests that the human teacher, the student and the technology can be intertwined. We asked students to engage with digital objects and the technology to enhance their learning journey. As teachers we do not merely support the digital learner but we remain at the centre of teaching the course.

[Fawns et al. \(2019\)](#) point out that online learning is a key growth area in higher education, which is even more true since the pandemic started, but that it is harder to form relationships in online courses. Therefore, we saw it as important to develop online dialogue between students in order to form communities which can improve these relationships. Building a community online can be harder but it is possible. We tried to achieve this through using a combination of traditional learning such as lectures

²<https://www.nltk.org>

³<https://spacy.io>

and task-based learning such as pair programming exercises. Online learning tends to be interrupted as we are in our homes or elsewhere and have responsibilities that can take us away from the online space, bandwidth issues, dropping children at school, flatmates interrupting, phone calls, even the door bell ringing. Our teaching practices needed to be accepting of and adapted to this context.

Ross et al. (2013) discuss the issues of presence and distance in online learning. Interruptions in students' concentration are a common event when learning online and we must use resilience strategies to maintain a 'nearness' to our students. This includes recognising that these events are normal and that engaging is an effort, identifying affinities and creating a socialness, valuing that distraction can change our perspective and this is helpful and designing openings, events that allow and encourage student to come together and engage. Whilst designing the course we kept these ideas in focus in order to allow us to develop and enhance our online relationships and our students' learning.

3 A Virtual Learning Experience

3.1 The Team

Our team is made up of three early career academics at the University of Edinburgh. Two teaching fellows have a background in Natural Language Processing with PhDs in Computational Linguistics. The third teaching fellow has a PhD in Computer Science and frequently teaches programming to different types of audiences, including business students as well as students outside of higher education. The author list of this paper also includes a fourth (last) author who was a participant of our first pilot, is a lecturer herself, and who has provided us with useful feedback for future iterations of this course (see Section 4.2).

3.2 Course Overview

In our data-driven society, it is increasingly essential for people throughout the private, public and third sectors to know how to analyse the wealth of information society creates each day. Our TDM course gives participants who have no or very limited coding experience the tools they need to interrogate data. This course is designed to teach non-coders how to analyse textual data using Python as the main programming language. It takes them through the required steps needed to be able to analyse and visualise information in large sets of

textual document collections, or corpora.

The course takes place over three three-hour sessions and each session introduces participants to a new topic through a short lecture. The topics build on the previous sessions and at the end of each session there is time for discussion and feedback. In the first session we start with Python for reading in and processing text and teach how individual documents are loaded and tokenised. We work with plain text files but do raise the issue that textual data can be stored in different formats. However, to keep things simple we do not cover other formats in detail in the practical sessions.

In the second session we show how this is done using much larger sets of text and add in visualisations. We used two data sets as examples, the Medical History of British India (of Scotland, 2019) made available by the National Library of Scotland⁴ and the inaugural addresses of all American Presidents from 1789 to 2017. We show how participants can create concordance lists, token frequency distributions in a corpus and over time as well as lexical dispersion plots and how they can perform regular expression searches using Python. In this session we also explain that textual data can be messy and that a lot of time can be spent on cleaning and preparing data in a way that is most useful for further analysis. For example, we point students at stop words and punctuation in the results and explain how to filter them when creating frequency-based visualisations.

During the third session we cover POS-tagging and named entity recognition. This last session concludes with a lesson on visualisations of text and derived data by means of text highlighting, frequency graphs, word clouds and networks (see some examples in Figure 1). The underlying NLP tools used for this course are NLTK 3 and spaCy which are widely used for NLP research and development. This is also where we put some of the course material in context of our own research to show how it can be applied in practice in a real project. For example, we mentioned our previous work on collecting topic-specific Twitter datasets for further analysis (Llewellyn et al., 2015), on geoparsing historical and literary text (Clifford et al., 2016; Alex et al., 2019a) and on named entity recognition for radiology reports (Alex et al., 2019b; Gorinski et al., 2019).

⁴<https://data.nls.uk/data/digitised-collections/a-medical-history-of-british-india/>

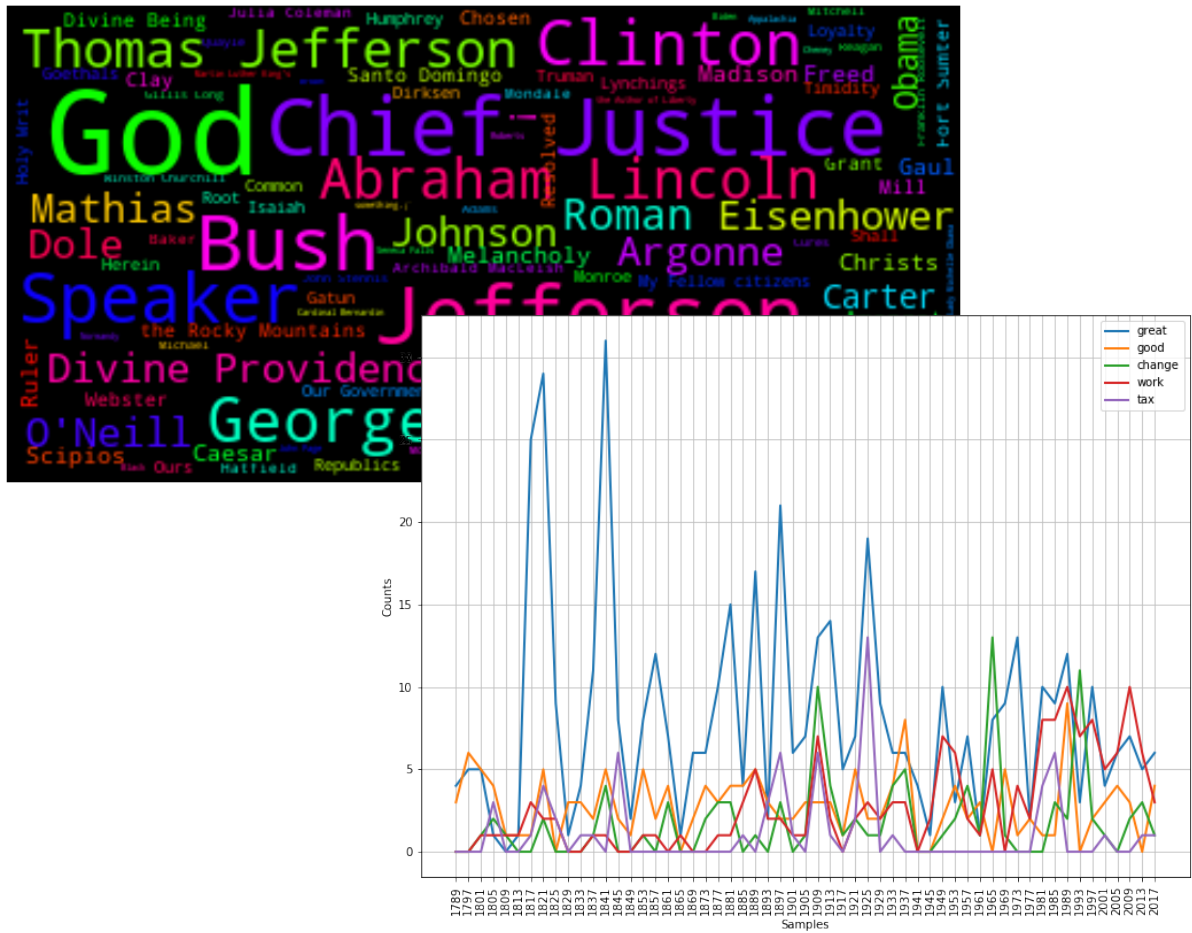


Figure 1: Visualisations of text explorations created by the students.

In the two pilots, we ran this course over three afternoon sessions on Monday, Wednesday and Friday, with an office hour on the days in-between to sort out any potential technical issues and answer questions. The main learning outcome is that by the end of the course the participants will have acquired initial TDM skills which they can use in their own research and build on by taking more advanced NLP courses or tutorials. A main goal of this course is to teach the material in a clear step-by-step way so all Python code and the examples are specific to each task but do not go in-depth into complicated programming concepts which we believe would confuse complete novices.

3.3 Pilot 1

In the first pilot we wanted to test the content of this course but also different methods for teaching online. We are all likely to be teaching virtually more often in the future even once the pandemic subsides. For example, EFI was planning to run hybrid courses to students across the world, even prior

to COVID-19. In this new world, we believe that online and hybrid teaching is here to stay alongside teaching students in the classroom. Higher education will need to determine their offer of different experiences to students be they on site or participating online synchronously or asynchronously.

We limited the first pilot to 25 participants. The backgrounds of students who signed up for our course were mixed coming from Law, Linguistics and Business. Everyone was either a student or a member of staff at the University of Edinburgh, where we had advertised the course, including every level from professor to undergraduate, joining from around the world. Some students even participated from different time zones.

On each day we started with a short presentation discussing the TDM theory of what was being taught in the practical session that followed. In the first pilot this was a live lecture, not recorded, allowing us to adapt the content to questions that came up during the course. When one teacher spoke the other two managed the video chat, answer-

ing questions or dealing with specific problems from students, and raising questions to the speaker. This was something we found was essential as it was very easy to lose flow and get distracted without this help. We learned then that it would have been extremely challenging to teach this course live online single-handedly and after each session expressed appreciation that there were three of us helping each other.

We used a variety of technologies provided by the university. Learn,⁵ our in-house virtual learning environment (VLE), was used to provide access to course materials. We met with students virtually using the Blackboard Collaborate software⁶ which is accessible through Learn. Aside from the video itself, we used text chat, the virtual whiteboard, polls, the ability to raise a hand, breakout groups, file sharing, and screen sharing, all functionalities which have become second nature after a year of pandemic but which when we ran the first pilot were for the most part still fairly unfamiliar to many participants. We also used Noteable,⁷ the University of Edinburgh's in-house notebook platform, to provide a virtual programming environment (VPE) with Jupyter Notebooks,⁸ and used GitHub⁹ to provide students access to the course material and code. We note that the students did not have to learn how to use GitHub, which would be a big ask for coding novices, but merely had to paste the GitHub link of the corresponding material into Noteable which then automatically loaded the material in the form of a notebook.

Each day the students were given two sets of worked through problems using the VPE which they used directly through the VPL in their own browser. We found this to be a really important tool for everyone as it reduced the need for students to download and set up software on different operating systems and alleviated us from doing a lot of technical support to get students set up and running for all the practical parts of the course.

During the sessions the students were given a link to a GitHub repository from which they could pull new notebooks onto the VPE at the beginning of each session. The notebooks include a combi-

nation of explanations, code to run and mini or extended programming tasks. For each approximately hour-long coding session students were assigned a random buddy and which they were put in a break-out room within the Collaborate video call. By now we are used to teaching and/or learning online and have likely experienced joining break-out rooms but at the time when we ran the first pilot most of our participants had never been in a break-out room before. So that experience took some getting used to. We described it as *feeling like being put in a separate room with your buddy. You can chat and share screens without being overheard by other people*. If the students got stuck on a particular coding problem or line of code and could not solve the issue together, they could raise a virtual hand and an instructor would drop into the room to help and answer questions or resolve programming issues. We also regularly popped into the rooms to see how everyone was doing, something which was well received by the students.

One of our team members is a strong proponent of pair programming (Williams et al., 2000; Hanks et al., 2011), where two students work together on a single machine to solve problems. This allows each pair of students to learn from each other as well as from their teacher(s) and thereby helps to broaden participation and to dispel the myth that programmers work on their own (Williams, 2006). We wanted to see if it was possible to take this approach into a virtual teaching environment. In addition to the students learning TDM skills, it also provided an opportunity for social interaction which was particularly welcome when we first piloted our course at the tail end of the first wave of COVID-19 in the UK and after weeks of strict lockdown with no or little opportunity to meet and interact with people outside one's own household.

One advantage of Blackboard Collaborate is that instructors are able to see visually when the people in break-out rooms are chatting to each other. This helped us to gauge if students embraced our pair programming experiment or if they preferred to work quietly "side-by-side" but connected virtually.

After each practical session we pulled everyone back into the shared room and asked participants to fill in a quick survey to give us feedback. We answered any questions, had a quick break, and then moved onto the next notebook with a new buddy. We wrapped up each session with a short Q&A and another round of feedback.

⁵<https://www.learn.ed.ac.uk>

⁶https://help.blackboard.com/Learn/Instructor/Interact/Blackboard_Collaborate

⁷<https://noteable.edina.ac.uk>

⁸<https://jupyter.org>

⁹<https://github.com>

3.4 Pilot 2

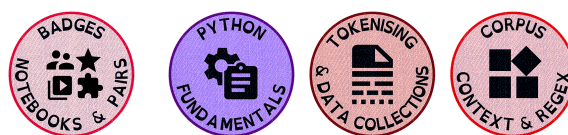
By the time of the second pilot in September 2020, we had gotten a lot more used to online meetings and two members of the teaching team had trained in a summer course on hybrid teaching called An Edinburgh Model for Teaching Online. This time we allowed 30 participants to sign up with over half of them from Scottish Government and the commercial sector, alongside university students and staff.

The main change we made to our first pilot, without altering the course content, is that we restructured the course material into teaching with digital badges (Gibson et al., 2015; Muilenburg and Berge, 2016) which are used in gamification of education (Dicheva et al., 2015; Ostaszewski and Reid, 2015). The principles that guided us were: flexibility, compartmentalisation and empowering the learner. Each badge is built around a Threshold concept (Land et al., 2005), a core step or skill (a ‘eureka’ moment) that opens the doors to further learning. Using a clear name and symbol, each badge signposts students’ takeaways and how it fits within the top level learning journey (see Figure 2).

The macro-structure in which badges form our course is complemented by a micro-structure of each badge: background theory and instructional content, code-along videos, notebooks with worked examples, exercises of increasing difficulty, relentless feedback, pair work and mini coding problems (with solutions). Badges build on top of each other, forming branches and enabling optional, further learning. Additionally, the modular micro-structure, enables easier switching between platforms or teaching modes (e.g. videos versus slides) and multiplies the benefits of improvements. Badges proved to be a promising format for delivering teaching of this course, especially in times of change, disruption and pivoting.

We wanted to give us and our course participants more flexibility, so we recorded all of the short lectures presented at the start of each badge and situated before each coding session in the course. This allowed students to come back to the recorded lecture materials later-on. It also gave us more flexibility answering questions in the chat, solving technical issues in the background and discussing the running of a given badge in a teaching team break-out room while participants were watching the video lecture.

BADGES DAY 1:



DAY 2:



DAY 3:



Figure 2: The badges used in our TDM course. We created them using Android Material Design Icons which are open source under Apache License 2.0.

4 Feedback

4.1 Relentless Feedback

In both pilots we collected relentless feedback. This feedback loop helped us to address questions raised and go over things that were unclear. We found it was really important to be flexible and adapt to what the students wanted. The twice-a-session mini-feedback form was really helpful for that and we made it very clear which parts of the course on day 2 and 3 were in response to participants’ feedback (see feedback analysis in Figure 3).

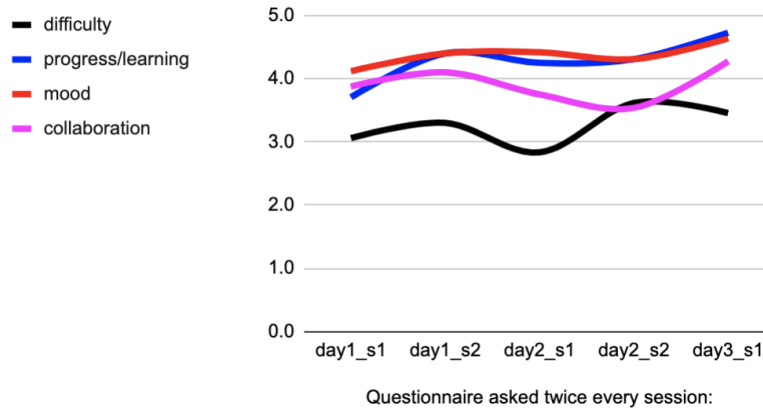
For example, a comments we received in the first pilot was that the students would prefer a quick recap of the previous session, which we then started doing and was a great way to link sessions and get the course material fresh in everyone’s minds. Given the feedback, we also worked through the first section of a notebook together, so everyone had a clear idea of what to do.

The relentless feedback and our response is one of the reasons we believe we had such a high participant retention rate which we were very pleased about. The pilots was free of charge, non-compulsory and ran over three afternoons. At least two thirds of the students who joined at the start of the week completed the last session on Friday.

We received constructive criticism but overall had very positive feedback on the course which, especially after the first pilot, made us feel very motivated having just had completed teaching our first online course. One participant thought it was “Fantastic!” in our final feedback survey. Another wrote “The pair learning is excellent! Jupiter [sic] notebooks are a great tool. The real-time interactivity is super rewarding.” Others reported that the

	AVG	MODE	day1_s1	day1_s2	day2_s1	day2_s2	day3_s1
difficulty	3.2	3	3.1	3.3	2.8	3.6	3.5
progress/learning	4.2	4	3.7	4.4	4.3	4.3	4.7
mood	4.3	5	4.1	4.4	4.4	4.3	4.6
collaboration	3.9	5	3.9	4.1	3.8	3.5	4.3

Learning KPIs across 3 days



“Fantastic! The pair learning is excellent! Jupiter notebooks are a great tool. The real time interactivity is super rewarding.”

Figure 3: Feedback analysis for all surveys over the course of the boot camp and a quote from one student (with permission to share). We asked students to record difficulty of the course, their progress and learning, their mood and how they felt about their collaboration in pairs as key performance indicators (KPIs) throughout the course.

lecturers and the “humour and playfulness of the examples” made the course “really great, especially for someone completely new to coding.” Yet another person commented that they would use the skills they learned in gathering data for their undergraduate dissertation about their research project.

4.2 Detailed Student Feedback

The following account is a more detailed reaction to our course provided by one of the student who participated in the first pilot of the TDM course and who we include as an author on this paper:

I was one of the mature students on the first pilot of the TDM Workshop – an academic myself with quantitative methods and coding experience in Stata and MatLab but not in Python, nor any previous experience with text mining or natural language processing.

I appreciated the feedback requests at the end of each session via Microsoft Office forms and the

immediate showcasing of the results for the whole class. Whenever there were bandwidth issues, the teaching team coordinated instantaneously and took over from each other.

The part of the course that taught me the most were the pair breakout rooms where we worked through computational Jupyter notebooks. The annotation of the exercises was invaluable, as were the videos showing one of the instructors working through a notebook themselves and importantly running into an error and explaining how we use the error message as guidance to fix the code. During the breakout sessions having the three instructors drop in and answer any questions was an excellent balance of allowing the students independence while also feeling supported. Working with different partners every time was also very valuable. When taking an active role and talking through the lines of code and my understanding of the outcome, I was able to check in with my partner and

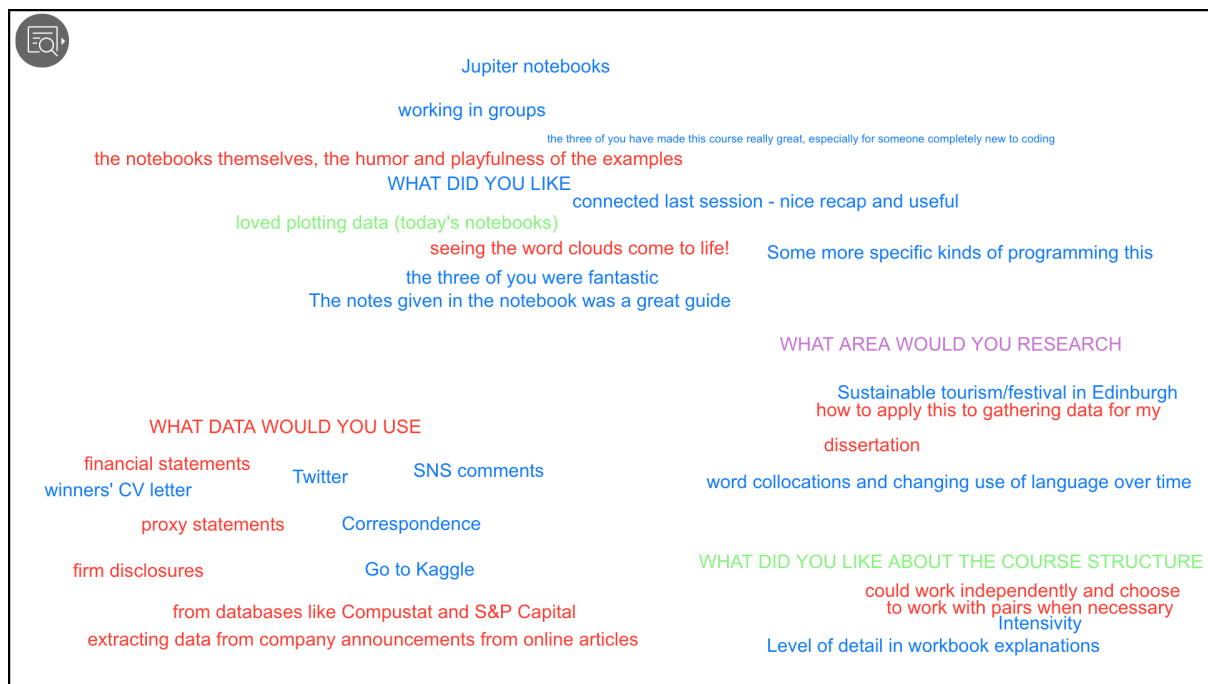


Figure 4: Whiteboard with feedback generated by students in the course.

be exposed to their style and approach to learning. Similarly, when taking the passive role and witnessing their way to working through a computational notebook, I could take away ideas of how to explain my thinking and understanding of the code in different ways.

The distribution of new material via GitHub was very efficient. The interactions via the virtual whiteboard created playfulness and joy in the learning process. Although I did not participate in the second pilot and was not exposed to the Badges, I see them as another element of enhancing the playfulness of the process.

The TDM Workshop I participated in took place relatively early in the pandemic before "Zoom fatigue" had set in and participants were excited to engage. A year later, full-time students appear to have become more resistant to engaging in voice and/or visual participation.

There were some points that required improvement, for example typos in the annotation of the computational notebooks or some time being eaten up by technical troubleshooting. However, even these created an atmosphere of immediacy, flexibility and a sense of "We are all in this together".

Overall, I benefited immensely from taking the first pilot. Not only do I now have an idea of text mining tools and how to use them but I was also inspired by and adopted the computational notebooks

in my own teaching of *Investments in the Autumn of 2020*. I also implemented regular feedback, which I felt provided the element of playfulness and joy, in an even more interactive platform with gifs, word-clouds and animations (using Mentimeter¹⁰).

5 Lessons Learned

Despite on-the-whole positive comments, we still found teaching in an online environment quite odd. We felt that we lost the sense of whether the students were engaged, learning and enjoying the experience because most participants had their cameras switched off so we could not see their faces or body language. The feedback did help, even simply asking students to 'raise your hand if you can hear me', but it still remains odd to us to talk to a blank screen without seeing everyone.

We did not get everything right. The technology did not always work but luckily one teaching team member is quite experienced in fixing software-related issues. We would have struggled without it. Initially we also did not give enough thought to accessibility; we just assumed the software would deal with that - it did not. We learned that we have to ask all students before the course if they might have issues in accessing course materials or video calls and make time to deal with any technical issues that could arise as a result.

¹⁰<https://www.mentimeter.com>

We learned that students can be shy when it comes to talking to each other and putting on their webcams. We found ice breaker questions upfront, can be answered playfully on a whiteboard, very helpful for putting students at ease and have some fun. We used some simple things that made a lot of difference. We played music in the room before the class so when students joined, they knew we were there and that their speakers were on. We made extensive use of the virtual whiteboard to gather anonymous feedback really fast in addition to frequent short surveys (see Figure 4). We also included questions in the notebooks that buddies had to work on together to encourage discussion. The notebooks contained essential TDM coding tasks and more complex tasks for the curious. This allowed some students to extend their learning without others feeling they were left behind.

We also found that the amount of content that we could cover grew as the course went on. There were initial issues with the technology which needed fixing and as we were all getting used to the new way of teaching. The conversation also became more natural as time went on. At first it was quite odd to drop into the break-out rooms but by the second session this became easier and we were all chatting a lot more. The majority of students really liked the pair programming, they liked the flexibility and the content. They really felt they were part of the course in a way that is not always experienced online.

As instructors, we found that teaching in this way, switching between modes, lecturing, answering the chat, live coding and responding to issues is really cognitively challenging. It is hard work and cannot easily be done by one individual. The technologies we use are complex and can fail but they are for the most part intuitive and provide a wide range of ways to teach and interact. We learned that online teaching is exhausting but done right it can still be really rewarding. We all enjoyed the interactions and felt part of a little community. After the course we did a debriefing and each wrote down three things we liked about the course and something we wished we could have achieved (see page Appendix on 11).

We, the TDM course teachers on this paper, have, in the same way as the author who participated in the course, benefited immensely from what we learned through these pilots before delving into our online teaching in the first term of 2020/21.

6 Summary and Future Work

In this paper, we have reflected of how we ported a TDM course online as a result of the global pandemic caused by COVID-19. We described the content of the course and how we adapted it over two pilot runs. We particularly found different features of Blackboard Collaborate useful for teaching, especially the use of a virtual whiteboard and dividing the class up into break-out rooms. Students responded positively to learning in pairs and to course materials broken down into digital badges. Finally, the relentless feedback we collected throughout each session and after the course helped us as teachers to improve the course and how we teach it. To make a course like this a good learning experience, it is really important to build community and get students to talk not just to the teachers but to each other as they would in a classroom.

Being caught in lockdown encouraged us to innovate, and our experience demonstrates what is possible to achieve virtually despite the limitations. Experiencing the learning in a classroom is difficult to replicate online, however, we are confident that these types of virtual environments will play a role in education beyond this pandemic, to complement and enhance traditional learning.

Going forward we would like to experiment with teaching this course in different ways: asynchronously to students joining from different time zones, to much larger groups to understand where the limits are in terms of number of participants given staff capacity, or in a writer-retreat type setup where the instructors touch base with students several times during the day. We will also look at how this course can be pivoted back to on-campus teaching for students who can join in person and once the current pandemic slows down, lockdown restrictions are relaxed and on-campus teaching resumes. We are pleased to announce that this course will be part of the post-graduate programme taught at EFI.

Acknowledgements

We thank Professor Laura Cram for supporting us in the long-term development of this course as part of EFI's PGT training programme. Our course was built on and expands material developed for a Library Carpentries course¹¹ with the support of the Centre for Data, Culture and Society.¹² We

¹¹<http://librarycarpentry.org/lc-tdm/>

¹²<https://www.cdcs.ed.ac.uk>

would also like to thank Siobhan Dunn and her colleagues at EFI for managing the registration for our courses and Marco Rossi at the University of Edinburgh Business School's Student Development Team for offering the course to their students.

References

- Apoorv Agarwal. 2013. [Teaching the basics of NLP and ML in an introductory course to information science](#). In *Proceedings of the Fourth Workshop on Teaching NLP and CL*, pages 77–84, Sofia, Bulgaria. Association for Computational Linguistics.
- Beatrice Alex, Claire Grover, Richard Tobin, and Jon Oberlander. 2019a. [Geoparsing historical and contemporary literary text set in the City of Edinburgh](#). *Language Resources and Evaluation*, 53(4):651–675.
- Beatrice Alex, Claire Grover, Richard Tobin, Cathie Sudlow, Grant Mair, and William Whiteley. 2019b. [Text mining brain imaging reports](#). *Journal of Biomedical Semantics*, 10(1):1–11.
- Sian Bayne. 2015. [Teacherbot: interventions in automated teaching](#). *Teaching in Higher Education*, 20(4):455–467.
- Steven Bird, Ewan Klein, and Edward Loper. 2005. [NLTK tutorial: Introduction to natural language processing](#). *Creative Commons Attribution*.
- Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: Analyzing text with the natural language toolkit*. "O'Reilly Media, Inc."
- Jim Clifford, Beatrice Alex, Colin M Coates, Ewan Klein, and Andrew Watson. 2016. [Geoparsing history: Locating commodities in ten million pages of nineteenth-century sources](#). *Historical Methods: A Journal of Quantitative and Interdisciplinary History*, 49(3):115–131.
- Darina Dicheva, Christo Dichev, Gennady Agre, and Galia Angelova. 2015. [Gamification in education: A systematic mapping study](#). *Journal of Educational Technology & Society*, 18(3):75–88.
- Tim Fawns, Gill Aitken, and Derek Jones. 2019. [Online learning as embodied, socially meaningful experience](#). *Postdigital Science and Education*, 1(2):293–297.
- David Gibson, Nathaniel Ostashewski, Kim Flintoff, Sheryl Grant, and Erin Knight. 2015. [Digital badges in education](#). *Education and Information Technologies*, 20(2):403–410.
- Philip John Gorinski, Honghan Wu, Claire Grover, Richard Tobin, Conn Talbot, Heather Whalley, Cathie Sudlow, William Whiteley, and Beatrice Alex. 2019. [Named entity recognition for electronic health records: A comparison of rule-based and machine learning approaches](#). ArXiv.
- Brian Hanks, Sue Fitzgerald, Renée McCauley, Laurie Murphy, and Carol Zander. 2011. [Pair programming in education: A literature review](#). *Computer Science Education*, 21(2):135–173.
- Marti A Hearst. 2005. [Teaching applied natural language processing: Triumphs and tribulations](#). In *Proceedings of the Second ACL Workshop on Effective Tools and Methodologies for Teaching NLP and CL*, pages 1–8.
- Dirk Hovy. 2020. *Text Analysis in Python for Social Scientists: Discovery and Exploration*. Cambridge University Press.
- David Jurgens and Lucy Li. 2018. [A Look Inside the Pedagogy of Natural Language Processing](#). 25/09/2018. Accessed on 23/04/2021.
- Ray Land, Glynis Cousin, Jan HF Meyer, and Peter Davies. 2005. [Threshold concepts and troublesome knowledge \(3\): implications for course design and evaluation](#). *Improving student learning diversity and inclusivity*, 4:53–64.
- Clare Llewellyn, Claire Grover, Beatrice Alex, Jon Oberlander, and Richard Tobin. 2015. [Extracting a topic specific dataset from a Twitter archive](#). In *International Conference on Theory and Practice of Digital Libraries*, pages 364–367. Springer.
- Barbara McGillivray, Beatrice Alex, Sarah Ames, Guyda Armstrong, David Beavan, Arianna Ciula, Giovanni Colavizza, James Cummings, David De Roure, Adam Farquhar, et al. 2020. [The challenges and prospects of the intersection of humanities and data science: A white paper from The Alan Turing Institute](#).
- Lin Y Muilenburg and Zane L Berge. 2016. *Digital badges in education: Trends, issues, and cases*. Routledge.
- National Library of Scotland. 2019. [A Medical History of British India](#). Accessed on 23/04/2021.
- Nathaniel Ostashewski and Doug Reid. 2015. [A history and frameworks of digital badges in education](#). In *Gamification in education and business*, pages 187–200. Springer.
- Jen Ross, Michael Sean Gallagher, and Hamish Macleod. 2013. [Making distance visible: Assembling nearness in an online distance learning programme](#). *International Review of Research in Open and Distributed Learning*, 14(4):51–67.
- Laurie Williams. 2006. [Debunking the nerd stereotype with pair programming](#). *Computer*, 39(5):83–85.
- Laurie Williams, Robert R Kessler, Ward Cunningham, and Ron Jeffries. 2000. [Strengthening the case for pair programming](#). *IEEE software*, 17(4):19–25.

A Appendix: Three Stars and a Wish

Clare:

- ★ I liked that we were all willing to try anything and go beyond our comfort zone and fail
- ★ I liked the way we naturally supported each other and took different roles, and swapped those roles
- ★ I enjoyed learning the technology – something I thought I'd hate!
- ✧ *I wish we could find a way to make the students more interactive, chat, turn their cameras on.*



Pawel:

- ★ Relentless feedback: asking 30s pulse-checking questionnaires twice a day; we used Office Forms.
- ★ Staff room: easy and persisting internal comms channel within the course development team; we used Teams.
- ★ Poetic licence: the person who created the final version of the notes was allowed to make adjustments to the content, so they fit the format.
- ✧ *Our greatest ally was the tech that just worked. I wish we could also provide core hybrid experience of being in the same room.*



Beatrice:

- ★ Great team of people, great combination of skills
- ★ Brilliant to see feedback from students coming in
- ★ I learned a lot about teaching online
- ✧ *I wish I could witness the learning better online.*

